

Mathmatial Foundations of Reinforcement Learning

Policy optimization: REINFORCE, PG and NPG



Yuting Wei

Statistics & Data Science, Wharton
University of Pennsylvania

Fall 2023

Outline

Introduction to policy gradient methods

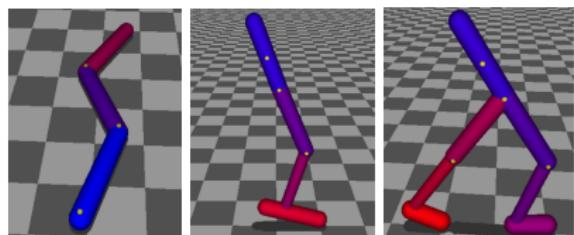
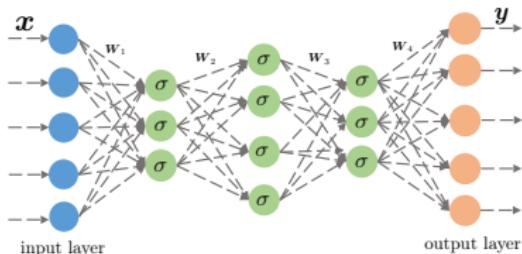
Global convergence of softmax policy gradient methods

Natural policy gradient methods

Policy optimization

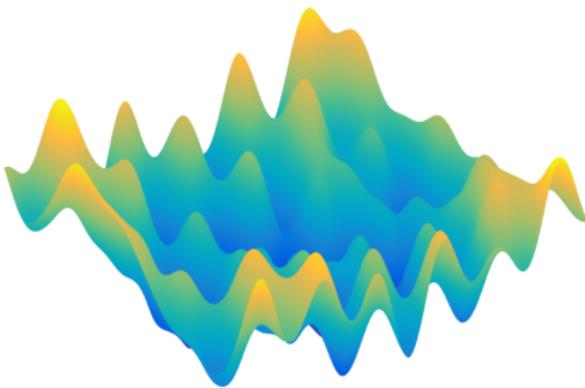
$$\text{maximize}_{\theta} \quad \text{value}(\text{policy}(\theta))$$

- directly optimize the policy, which is the quantity of interest;
- allow flexible differentiable parameterizations of the policy;
- work with both continuous and discrete problems.



Theoretical challenges: non-concavity

Little understanding on the global convergence of policy gradient methods until very recently, e.g. (Fazel et al., 2018; Bhandari and Russo, 2019; Agarwal et al., 2019; Mei et al. 2020), and many more.



Our goal:

- introduce the algorithmic framework of popular policy gradient methods
- understand finite-time convergence rates of popular heuristics

Introduction to policy gradient methods

Policy gradient methods

Given an initial state distribution $s \sim \rho$, find policy π such that

$$\text{maximize}_{\pi} \quad V^{\pi}(\rho) := \mathbb{E}_{s \sim \rho} [V^{\pi}(s)]$$

Policy gradient methods

Given an initial state distribution $s \sim \rho$, find policy π such that

$$\text{maximize}_{\pi} \quad V^{\pi}(\rho) := \mathbb{E}_{s \sim \rho} [V^{\pi}(s)]$$



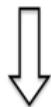
Parameterization:

$$\pi := \pi_{\theta}$$

Policy gradient methods

Given an initial state distribution $s \sim \rho$, find policy π such that

$$\text{maximize}_{\pi} \quad V^{\pi}(\rho) := \mathbb{E}_{s \sim \rho} [V^{\pi}(s)]$$



Parameterization:

$$\pi := \pi_{\theta}$$

$$\text{maximize}_{\theta} \quad V^{\pi_{\theta}}(\rho) := \mathbb{E}_{s \sim \rho} [V^{\pi_{\theta}}(s)]$$

Policy gradient methods

Given an initial state distribution $s \sim \rho$, find policy π such that

$$\text{maximize}_{\pi} \quad V^{\pi}(\rho) := \mathbb{E}_{s \sim \rho} [V^{\pi}(s)]$$



Parameterization:

$$\pi := \pi_{\theta}$$

$$\text{maximize}_{\theta} \quad V^{\pi_{\theta}}(\rho) := \mathbb{E}_{s \sim \rho} [V^{\pi_{\theta}}(s)]$$

Policy gradient method

For $t = 0, 1, \dots$

$$\theta^{(t+1)} = \theta^{(t)} + \eta \nabla_{\theta} V^{\pi_{\theta}^{(t)}}(\rho)$$

where η is the learning rate.

Policy gradient methods

Given an initial state distribution $s \sim \rho$, find policy π such that

$$\text{maximize}_{\pi} \quad V^{\pi}(\rho) := \mathbb{E}_{s \sim \rho} [V^{\pi}(s)]$$



Parameterization:

$$\pi := \pi_{\theta}$$

$$\text{maximize}_{\theta} \quad V^{\pi_{\theta}}(\rho) := \mathbb{E}_{s \sim \rho} [V^{\pi_{\theta}}(s)]$$

Policy gradient method

For $t = 0, 1, \dots$

$$\theta^{(t+1)} = \theta^{(t)} + \eta \nabla_{\theta} V^{\pi_{\theta}^{(t)}}(\rho)$$

where η is the learning rate.

How to calculate the gradient?

Policy gradient derivation

- Assume π_θ is differentiable when it is non-zero with gradient $\nabla_\theta \pi_\theta$.

Policy gradient derivation

- Assume π_θ is differentiable when it is non-zero with gradient $\nabla_\theta \pi_\theta$.

The policy gradient can be decomposed as

$$\begin{aligned}\nabla_\theta V^{\pi_\theta}(\rho) &= \nabla_\theta \mathbb{E}_{s_0 \sim \rho} [V^{\pi_\theta}(s_0)] \\ &= \mathbb{E}_{s_0 \sim \rho} \left[\nabla_\theta \left(\sum_{a_0 \in \mathcal{A}} \pi_\theta(a_0 | s_0) Q^{\pi_\theta}(s_0, a_0) \right) \right] \\ &= \mathbb{E}_{s_0 \sim \rho} \left[\sum_{a_0 \in \mathcal{A}} \nabla_\theta \pi_\theta(a_0 | s_0) Q^{\pi_\theta}(s_0, a_0) + \sum_{a_0 \in \mathcal{A}} \pi_\theta(a_0 | s_0) \nabla_\theta Q^{\pi_\theta}(s_0, a_0) \right]\end{aligned}$$

Policy gradient derivation

- Assume π_θ is differentiable when it is non-zero with gradient $\nabla_\theta \pi_\theta$.

The policy gradient can be decomposed as

$$\begin{aligned}\nabla_\theta V^{\pi_\theta}(\rho) &= \nabla_\theta \mathbb{E}_{s_0 \sim \rho} [V^{\pi_\theta}(s_0)] \\ &= \mathbb{E}_{s_0 \sim \rho} \left[\nabla_\theta \left(\sum_{a_0 \in \mathcal{A}} \pi_\theta(a_0 | s_0) Q^{\pi_\theta}(s_0, a_0) \right) \right] \\ &= \mathbb{E}_{s_0 \sim \rho} \left[\sum_{a_0 \in \mathcal{A}} \nabla_\theta \pi_\theta(a_0 | s_0) Q^{\pi_\theta}(s_0, a_0) + \sum_{a_0 \in \mathcal{A}} \pi_\theta(a_0 | s_0) \nabla_\theta Q^{\pi_\theta}(s_0, a_0) \right]\end{aligned}$$

We discuss the two terms separately.

Policy gradient derivation - first term

Note that

$$\nabla_{\theta} \pi_{\theta}(a|s) = \pi_{\theta}(a|s) \frac{\nabla_{\theta} \pi_{\theta}(a|s)}{\pi_{\theta}(a|s)} = \pi_{\theta}(a|s) \underbrace{\nabla_{\theta} \log \pi_{\theta}(a|s)}_{\text{score function}}.$$

Policy gradient derivation - first term

Note that

$$\nabla_{\theta} \pi_{\theta}(a|s) = \pi_{\theta}(a|s) \frac{\nabla_{\theta} \pi_{\theta}(a|s)}{\pi_{\theta}(a|s)} = \pi_{\theta}(a|s) \underbrace{\nabla_{\theta} \log \pi_{\theta}(a|s)}_{\text{score function}}.$$

The first term in the policy gradient is expressed as

$$\begin{aligned} & \mathbb{E}_{s_0 \sim \rho} \left[\sum_{a_0 \in \mathcal{A}} \nabla_{\theta} \pi_{\theta}(a_0|s_0) Q^{\pi_{\theta}}(s_0, a_0) \right] \\ &= \mathbb{E}_{s_0 \sim \rho} \left[\sum_{a_0 \in \mathcal{A}} \pi_{\theta}(a_0|s_0) \nabla_{\theta} \log \pi_{\theta}(a_0|s_0) Q^{\pi_{\theta}}(s_0, a_0) \right] \\ &= \mathbb{E}_{s_0 \sim \rho, a_0 \sim \pi_{\theta}(\cdot|s_0)} [\nabla_{\theta} \log \pi_{\theta}(a_0|s_0) Q^{\pi_{\theta}}(s_0, a_0)] \end{aligned}$$

Policy gradient derivation - second term

The second term in the policy gradient is expressed as

$$\begin{aligned} & \mathbb{E}_{s_0 \sim \rho} \left[\sum_{a_0 \in \mathcal{A}} \pi_\theta(a_0 | s_0) \nabla_\theta Q^{\pi_\theta}(s_0, a_0) \right] \\ &= \mathbb{E}_{s_0 \sim \rho, a_0 \sim \pi_\theta(\cdot | s_0)} [\nabla_\theta Q^{\pi_\theta}(s_0, a_0)] \\ &= \mathbb{E}_{s_0 \sim \rho, a_0 \sim \pi_\theta(\cdot | s_0)} [\nabla_\theta (r(s_0, a_0) + \gamma \mathbb{E}_{s_1 \sim P(\cdot | s_0, a_0)} V^{\pi_\theta}(s_1))] \\ &= \gamma \mathbb{E}_{s_0 \sim \rho, a_0 \sim \pi_\theta(\cdot | s_0), s_1 \sim P(\cdot | s_0, a_0)} [\nabla_\theta V^{\pi_\theta}(s_1)], \end{aligned}$$

which is similar to what we want to bound, but a discounted one-step look-ahead.

Policy gradient derivation - recursion

Letting τ denote the trajectory following policy π_θ , by recursion,

$$\begin{aligned}\nabla_\theta V^{\pi_\theta}(\rho) &= \mathbb{E}_{s_0 \sim \rho, a_0 \sim \pi_\theta(\cdot | s_0)} [\nabla_\theta \log \pi_\theta(a_0 | s_0) Q^{\pi_\theta}(s_0, a_0)] \\ &\quad + \gamma \mathbb{E}_{s_0 \sim \rho, a_0 \sim \pi_\theta(\cdot | s_0), s_1 \sim P(\cdot | s_0, a_0)} [\nabla_\theta V^{\pi_\theta}(s_1)]\end{aligned}$$

Policy gradient derivation - recursion

Letting τ denote the trajectory following policy π_θ , by recursion,

$$\begin{aligned}\nabla_\theta V^{\pi_\theta}(\rho) &= \mathbb{E}_{s_0 \sim \rho, a_0 \sim \pi_\theta(\cdot|s_0)} [\nabla_\theta \log \pi_\theta(a_0|s_0) Q^{\pi_\theta}(s_0, a_0)] \\ &\quad + \gamma \mathbb{E}_{s_0 \sim \rho, a_0 \sim \pi_\theta(\cdot|s_0), s_1 \sim P(\cdot|s_0, a_0)} [\nabla_\theta V^{\pi_\theta}(s_1)] \\ &= \mathbb{E}_{(s_0, a_0) \sim \tau} [\nabla_\theta \log \pi_\theta(a_0|s_0) Q^{\pi_\theta}(s_0, a_0)] \\ &\quad + \gamma \mathbb{E}_{(s_0, a_0, s_1, a_1) \sim \tau} [\nabla_\theta \log \pi_\theta(a_1|s_1) Q^{\pi_\theta}(s_1, a_1)] + \dots\end{aligned}$$

Policy gradient derivation - recursion

Letting τ denote the trajectory following policy π_θ , by recursion,

$$\begin{aligned}\nabla_\theta V^{\pi_\theta}(\rho) &= \mathbb{E}_{s_0 \sim \rho, a_0 \sim \pi_\theta(\cdot|s_0)} [\nabla_\theta \log \pi_\theta(a_0|s_0) Q^{\pi_\theta}(s_0, a_0)] \\ &\quad + \gamma \mathbb{E}_{s_0 \sim \rho, a_0 \sim \pi_\theta(\cdot|s_0), s_1 \sim P(\cdot|s_0, a_0)} [\nabla_\theta V^{\pi_\theta}(s_1)] \\ &= \mathbb{E}_{(s_0, a_0) \sim \tau} [\nabla_\theta \log \pi_\theta(a_0|s_0) Q^{\pi_\theta}(s_0, a_0)] \\ &\quad + \gamma \mathbb{E}_{(s_0, a_0, s_1, a_1) \sim \tau} [\nabla_\theta \log \pi_\theta(a_1|s_1) Q^{\pi_\theta}(s_1, a_1)] + \dots \\ &= \mathbb{E}_{(s_i, a_i)_{i \geq 0} \sim \tau} \left[\sum_{i=0}^{\infty} \gamma^i \nabla_\theta \log \pi_\theta(a_i|s_i) Q^{\pi_\theta}(s_i, a_i) \right]\end{aligned}$$

Policy gradient derivation - recursion

Letting τ denote the trajectory following policy π_θ , by recursion,

$$\begin{aligned}\nabla_\theta V^{\pi_\theta}(\rho) &= \mathbb{E}_{s_0 \sim \rho, a_0 \sim \pi_\theta(\cdot | s_0)} [\nabla_\theta \log \pi_\theta(a_0 | s_0) Q^{\pi_\theta}(s_0, a_0)] \\ &\quad + \gamma \mathbb{E}_{s_0 \sim \rho, a_0 \sim \pi_\theta(\cdot | s_0), s_1 \sim P(\cdot | s_0, a_0)} [\nabla_\theta V^{\pi_\theta}(s_1)] \\ &= \mathbb{E}_{(s_0, a_0) \sim \tau} [\nabla_\theta \log \pi_\theta(a_0 | s_0) Q^{\pi_\theta}(s_0, a_0)] \\ &\quad + \gamma \mathbb{E}_{(s_0, a_0, s_1, a_1) \sim \tau} [\nabla_\theta \log \pi_\theta(a_1 | s_1) Q^{\pi_\theta}(s_1, a_1)] + \dots \\ &= \mathbb{E}_{(s_i, a_i)_{i \geq 0} \sim \tau} \left[\sum_{i=0}^{\infty} \gamma^i \nabla_\theta \log \pi_\theta(a_i | s_i) Q^{\pi_\theta}(s_i, a_i) \right] \\ &= \frac{1}{1-\gamma} \mathbb{E}_{s \sim d_\rho^{\pi_\theta}, a \sim \pi_\theta(\cdot | s)} [Q^{\pi_\theta}(s, a) \nabla \log \pi_\theta(a | s)],\end{aligned}$$

where $d_\rho^{\pi_\theta}$ is the **state visitation distribution**:

$$d_{s_0}^\pi(s) = (1-\gamma) \sum_{t=0}^{\infty} \gamma^t \mathbb{P}^\pi(s_t = s | s_0), \quad d_\rho^\pi = \mathbb{E}_{s_0 \sim \rho} [d_{s_0}^\pi(s)].$$

The policy gradient theorem

Theorem 1 (Policy gradient theorem [Sutton et al., 1999])

The policy gradient can be evaluated via

$$\nabla_{\theta} V^{\pi_{\theta}}(\rho) = \frac{1}{1-\gamma} \mathbb{E}_{s \sim d_{\rho}^{\pi_{\theta}}, a \sim \pi_{\theta}(\cdot|s)} \left[Q^{\pi_{\theta}}(s, a) \nabla \log \pi_{\theta}(a|s) \right],$$

where

- $d_{\rho}^{\pi_{\theta}}$ is the state visitation distribution,
- $\nabla \log \pi_{\theta}(a|s)$ is the score function.

Provides an effective scheme for policy gradient evaluation (e.g., REINFORCE):

- rolling out trajectory following π_{θ}
- evaluating the value function $Q^{\pi_{\theta}}$

Examples of policy parameterization

Discrete action space: softmax parameterization with function approximation

$$\pi_\theta(a|s) \propto \exp(\phi(s, a)^\top \theta)$$

- $\phi(s, a)$ is the feature vector of each state-action pair;
- the score function $\nabla \log \pi_\theta(a|s) = \phi(s, a) - \mathbb{E}_{a \sim \pi_\theta(\cdot|s)}[\phi(s, \cdot)]$.

Examples of policy parameterization

Discrete action space: softmax parameterization with function approximation

$$\pi_\theta(a|s) \propto \exp(\phi(s, a)^\top \theta)$$

- $\phi(s, a)$ is the feature vector of each state-action pair;
- the score function $\nabla \log \pi_\theta(a|s) = \phi(s, a) - \mathbb{E}_{a \sim \pi_\theta(\cdot|s)}[\phi(s, \cdot)]$.

Continuous action space: Gaussian policy

$$a \sim \mathcal{N}(\mu(s), \sigma^2), \quad \mu(s) = \phi(s)^\top \theta$$

- $\phi(s)$ is the feature of each state;
- σ^2 is the variance (kept constant for simplicity);
- the score function $\nabla \log \pi_\theta(a|s) = \frac{(a - \mu(s))\phi(s)}{\sigma^2}$.

Baseline

The policy gradient can have high variance with limited samples.

Variance reduction: introducing a baseline

$$\nabla_{\theta} V^{\pi_{\theta}}(\rho) = \frac{1}{1 - \gamma} \mathbb{E}_{s \sim d_{\rho}^{\pi_{\theta}}, a \sim \pi_{\theta}(\cdot|s)} \left[(Q^{\pi_{\theta}}(s, a) - b(s)) \nabla \log \pi_{\theta}(a|s) \right],$$

to help minimize the variance:

$$\begin{aligned} \mathbb{E}_{a \sim \pi_{\theta}(\cdot|s)} \left[\nabla \log \pi_{\theta}(a|s) \right] &= \sum_a \pi_{\theta}(a|s) \frac{\nabla_{\theta} \pi_{\theta}(a|s)}{\pi_{\theta}(a|s)} \\ &= \sum_a \nabla_{\theta} \pi_{\theta}(a|s) \\ &= \nabla_{\theta} \sum_a \pi_{\theta}(a|s) = 0 \end{aligned}$$

Baseline

Variance reduction: introducing a baseline

$$\nabla_{\theta} V^{\pi_{\theta}}(\rho) = \frac{1}{1-\gamma} \mathbb{E}_{s \sim d_{\rho}^{\pi_{\theta}}, a \sim \pi_{\theta}(\cdot|s)} \left[(Q^{\pi_{\theta}}(s, a) - b(s)) \nabla \log \pi_{\theta}(a|s) \right],$$

to help minimize the variance.

- In practice, choose $b(s) = V^{\pi_{\theta}}(s)$, leading to

$$\nabla_{\theta} V^{\pi_{\theta}}(\rho) = \frac{1}{1-\gamma} \mathbb{E}_{s \sim d_{\rho}^{\pi_{\theta}}, a \sim \pi_{\theta}(\cdot|s)} \left[A^{\pi_{\theta}}(s, a) \nabla \log \pi_{\theta}(a|s) \right]$$

Baseline

Variance reduction: introducing a baseline

$$\nabla_{\theta} V^{\pi_{\theta}}(\rho) = \frac{1}{1-\gamma} \mathbb{E}_{s \sim d_{\rho}^{\pi_{\theta}}, a \sim \pi_{\theta}(\cdot|s)} \left[(Q^{\pi_{\theta}}(s, a) - b(s)) \nabla \log \pi_{\theta}(a|s) \right],$$

to help minimize the variance.

- In practice, choose $b(s) = V^{\pi_{\theta}}(s)$, leading to

$$\nabla_{\theta} V^{\pi_{\theta}}(\rho) = \frac{1}{1-\gamma} \mathbb{E}_{s \sim d_{\rho}^{\pi_{\theta}}, a \sim \pi_{\theta}(\cdot|s)} \left[A^{\pi_{\theta}}(s, a) \nabla \log \pi_{\theta}(a|s) \right]$$

- $A^{\pi}(s, a) = Q^{\pi}(s, a) - V^{\pi}(s)$ is the **advantage function**.

Baseline

Variance reduction: introducing a baseline

$$\nabla_{\theta} V^{\pi_{\theta}}(\rho) = \frac{1}{1-\gamma} \mathbb{E}_{s \sim d_{\rho}^{\pi_{\theta}}, a \sim \pi_{\theta}(\cdot|s)} \left[(Q^{\pi_{\theta}}(s, a) - b(s)) \nabla \log \pi_{\theta}(a|s) \right],$$

to help minimize the variance.

- In practice, choose $b(s) = V^{\pi_{\theta}}(s)$, leading to

$$\nabla_{\theta} V^{\pi_{\theta}}(\rho) = \frac{1}{1-\gamma} \mathbb{E}_{s \sim d_{\rho}^{\pi_{\theta}}, a \sim \pi_{\theta}(\cdot|s)} \left[A^{\pi_{\theta}}(s, a) \nabla \log \pi_{\theta}(a|s) \right]$$

- $A^{\pi}(s, a) = Q^{\pi}(s, a) - V^{\pi}(s)$ is the **advantage function**.
- Instead of estimating $Q^{\pi}(s, a)$, directly estimate $A^{\pi}(s, a)$.

Global convergence of softmax policy gradient methods

Softmax policy gradient methods

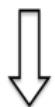
Given an initial state distribution $s \sim \rho$, find policy π such that

$$\text{maximize}_{\pi} \quad V^{\pi}(\rho) := \mathbb{E}_{s \sim \rho} [V^{\pi}(s)]$$

Softmax policy gradient methods

Given an initial state distribution $s \sim \rho$, find policy π such that

$$\text{maximize}_{\pi} \quad V^{\pi}(\rho) := \mathbb{E}_{s \sim \rho} [V^{\pi}(s)]$$



softmax parameterization:

$$\pi_{\theta}(a|s) \propto \exp(\theta(s, a))$$

Softmax policy gradient methods

Given an initial state distribution $s \sim \rho$, find policy π such that

$$\text{maximize}_{\pi} \quad V^{\pi}(\rho) := \mathbb{E}_{s \sim \rho} [V^{\pi}(s)]$$



softmax parameterization:

$$\pi_{\theta}(a|s) \propto \exp(\theta(s, a))$$

$$\text{maximize}_{\theta} \quad V^{\pi_{\theta}}(\rho) := \mathbb{E}_{s \sim \rho} [V^{\pi_{\theta}}(s)]$$

Softmax policy gradient methods

Given an initial state distribution $s \sim \rho$, find policy π such that

$$\text{maximize}_{\pi} \quad V^{\pi}(\rho) := \mathbb{E}_{s \sim \rho} [V^{\pi}(s)]$$



softmax parameterization:

$$\pi_{\theta}(a|s) \propto \exp(\theta(s, a))$$

$$\text{maximize}_{\theta} \quad V^{\pi_{\theta}}(\rho) := \mathbb{E}_{s \sim \rho} [V^{\pi_{\theta}}(s)]$$

Policy gradient method

For $t = 0, 1, \dots$

$$\theta^{(t+1)} = \theta^{(t)} + \eta \nabla_{\theta} V^{\pi_{\theta}^{(t)}}(\rho)$$

where η is the learning rate.

Global convergence of the PG method

Exact gradient evaluation: suppose we can perfectly evaluate the gradient

$$\nabla_{\theta} V^{\pi_{\theta}^{(t)}}(\rho),$$

does softmax policy gradient converge?

Global convergence of the PG method

Exact gradient evaluation: suppose we can perfectly evaluate the gradient

$$\nabla_{\theta} V^{\pi_{\theta}^{(t)}}(\rho),$$

does softmax policy gradient converge?

Theorem 2 ([Agarwal et al., 2021])

Assume ρ is strictly positive, i.e., $\rho(s) > 0$ for all states s . For $\eta \leq (1 - \gamma)^3/8$, then we have that for all states s ,

$$V^{(t)}(s) = V^{\pi_{\theta}^{(t)}}(s) \rightarrow V^*(s), \quad t \rightarrow \infty.$$

Global convergence of the PG method

Exact gradient evaluation: suppose we can perfectly evaluate the gradient

$$\nabla_{\theta} V^{\pi_{\theta}^{(t)}}(\rho),$$

does softmax policy gradient converge?

Theorem 2 ([Agarwal et al., 2021])

Assume ρ is strictly positive, i.e., $\rho(s) > 0$ for all states s . For $\eta \leq (1 - \gamma)^3 / 8$, then we have that for all states s ,

$$V^{(t)}(s) = V^{\pi_{\theta}^{(t)}}(s) \rightarrow V^*(s), \quad t \rightarrow \infty.$$

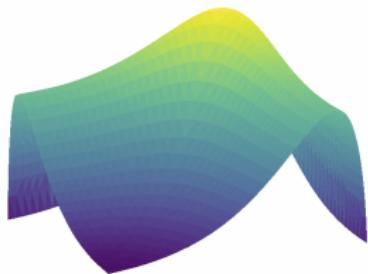
- Softmax policy gradient finds the global optimal policy despite concavity!

How fast does softmax PG converge?



- [Agarwal et al., 2021] showed that softmax PG converges **asymptotically** to the global optimal policy.

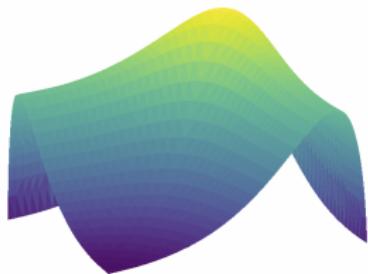
How fast does softmax PG converge?



- [Agarwal et al., 2021] showed that softmax PG converges **asymptotically** to the global optimal policy.
- [Mei et al., 2020] showed that softmax PG converges to global opt in

$$O\left(\frac{1}{\varepsilon}\right) \text{ iterations}$$

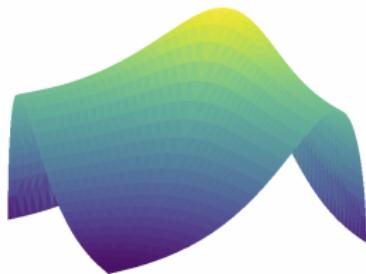
How fast does softmax PG converge?



- [Agarwal et al., 2021] showed that softmax PG converges **asymptotically** to the global optimal policy.
- [Mei et al., 2020] showed that softmax PG converges to global opt in

$$c(|\mathcal{S}|, |\mathcal{A}|, \frac{1}{1-\gamma}, \dots) O(\frac{1}{\varepsilon}) \text{ iterations}$$

How fast does softmax PG converge?



- [Agarwal et al., 2021] showed that softmax PG converges **asymptotically** to the global optimal policy.
- [Mei et al., 2020] showed that softmax PG converges to global opt in

$$c(|\mathcal{S}|, |\mathcal{A}|, \frac{1}{1-\gamma}, \dots) O(\frac{1}{\varepsilon}) \text{ iterations}$$

Is the rate of PG good, bad or ugly?

A negative message

Theorem 3 ([Li et al., 2023])

There exists an MDP s.t. it takes softmax PG at least

$$\frac{1}{\eta} |\mathcal{S}|^{2^{\Theta(\frac{1}{1-\gamma})}} \text{ iterations}$$

to achieve $\|V^{(t)} - V^\|_\infty \leq 0.15$.*

A negative message

Theorem 3 ([Li et al., 2023])

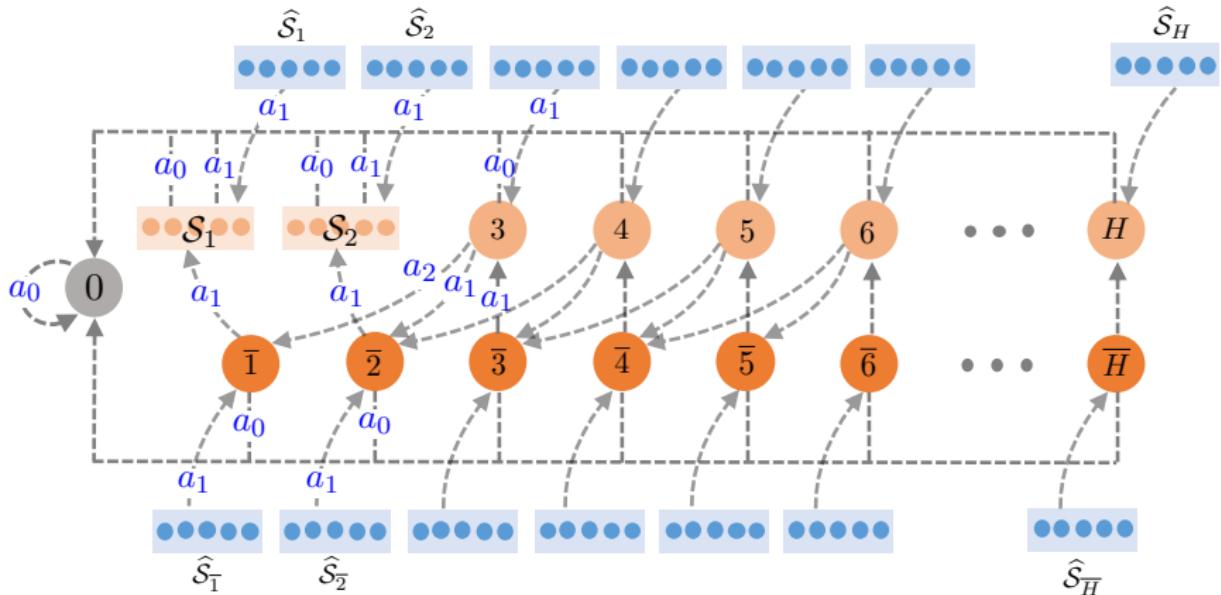
There exists an MDP s.t. it takes softmax PG at least

$$\frac{1}{\eta} |\mathcal{S}|^{2^{\Theta(\frac{1}{1-\gamma})}} \text{ iterations}$$

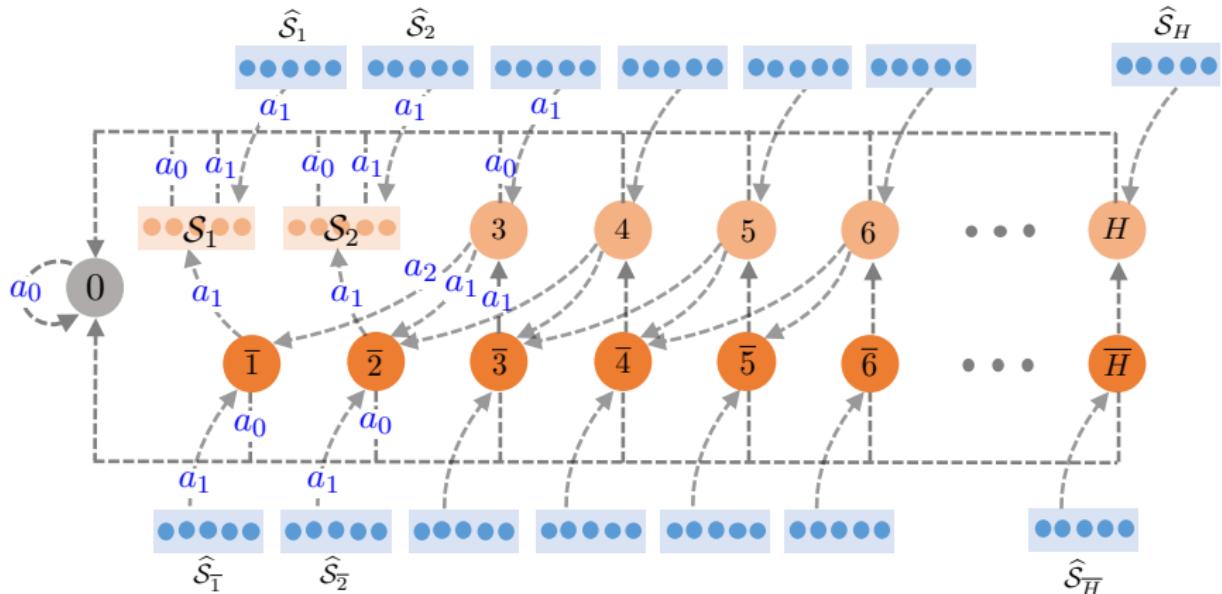
to achieve $\|V^{(t)} - V^*\|_\infty \leq 0.15$.

- Softmax PG can take (super)-exponential time to converge (in problems w/ large state space & long effective horizon)!
- Also hold for average sub-opt gap $\frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} [V^{(t)}(s) - V^*(s)]$.

MDP construction for our lower bound

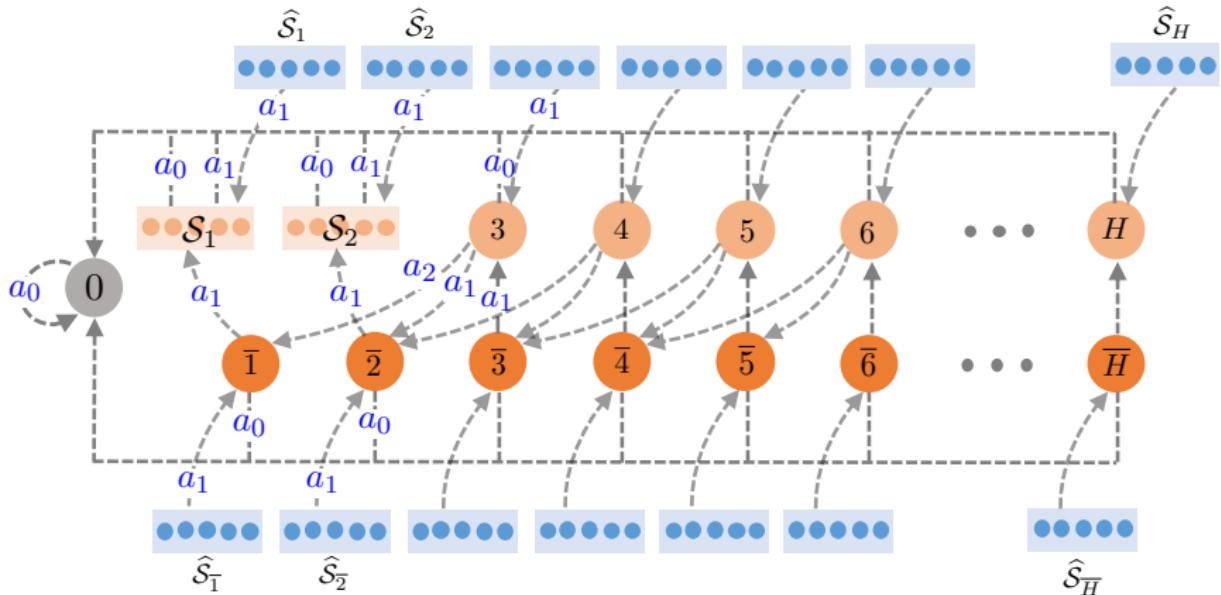


MDP construction for our lower bound



Key ingredients: for $3 \leq s \leq H \asymp \frac{1}{1-\gamma}$,

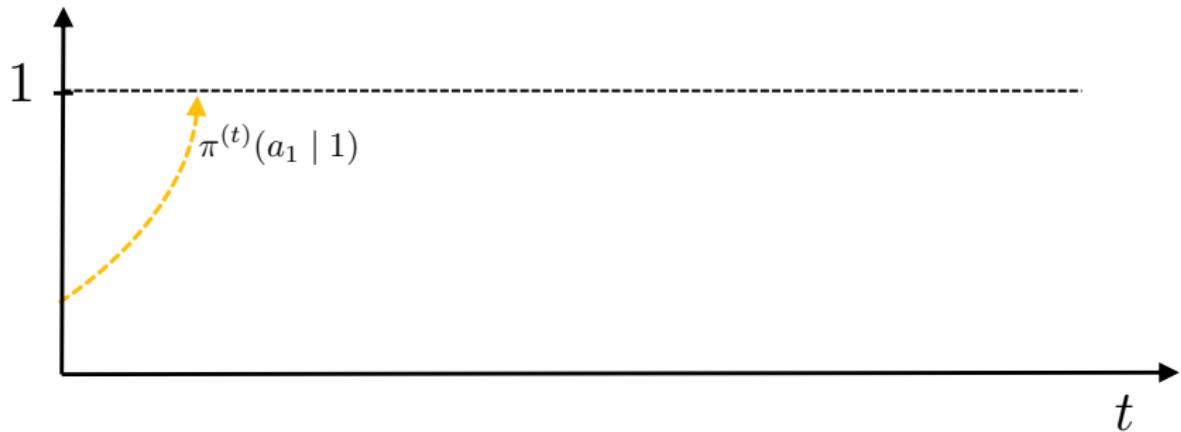
MDP construction for our lower bound



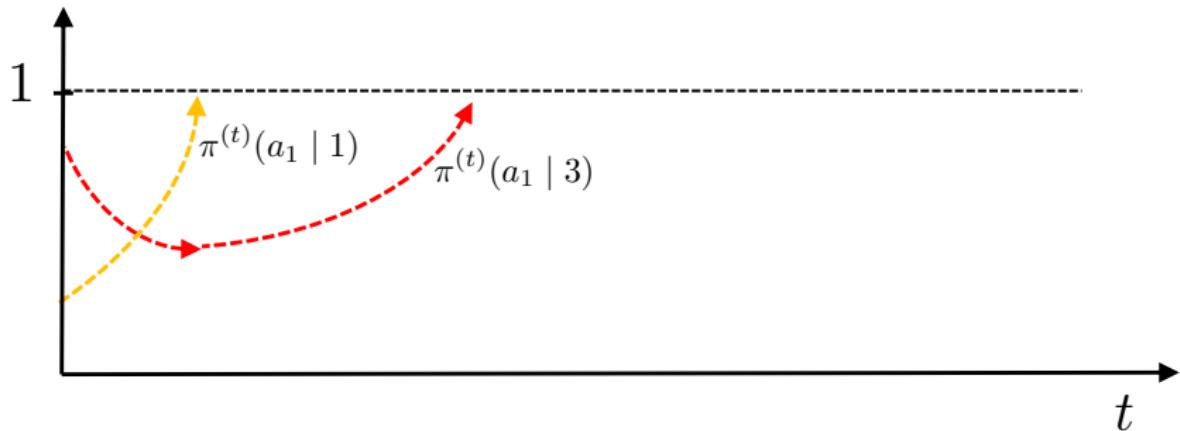
Key ingredients: for $3 \leq s \leq H \asymp \frac{1}{1-\gamma}$,

- $\pi^{(t)}(a_{\text{opt}} | s)$ keeps decreasing until $\pi^{(t)}(a_{\text{opt}} | s - 2) \approx 1$

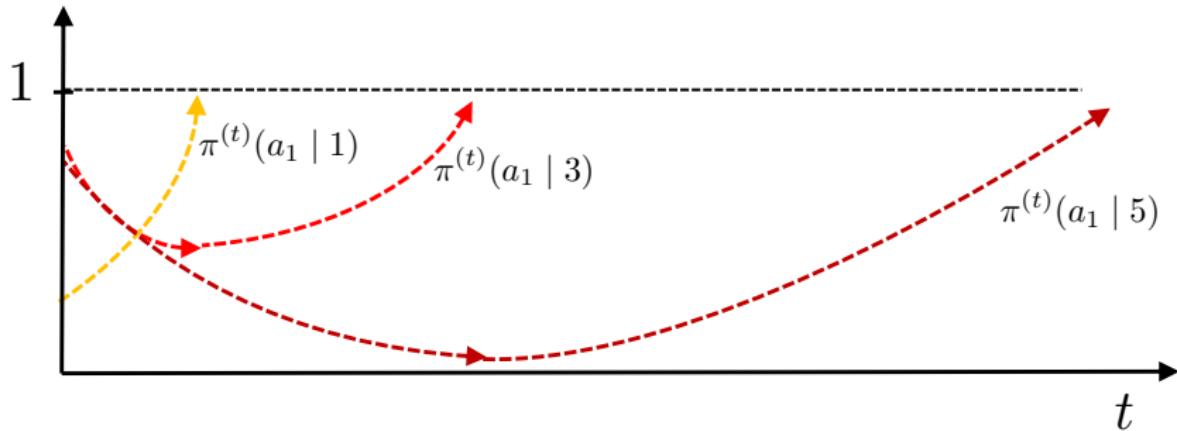
What is happening in our constructed MDP?



What is happening in our constructed MDP?

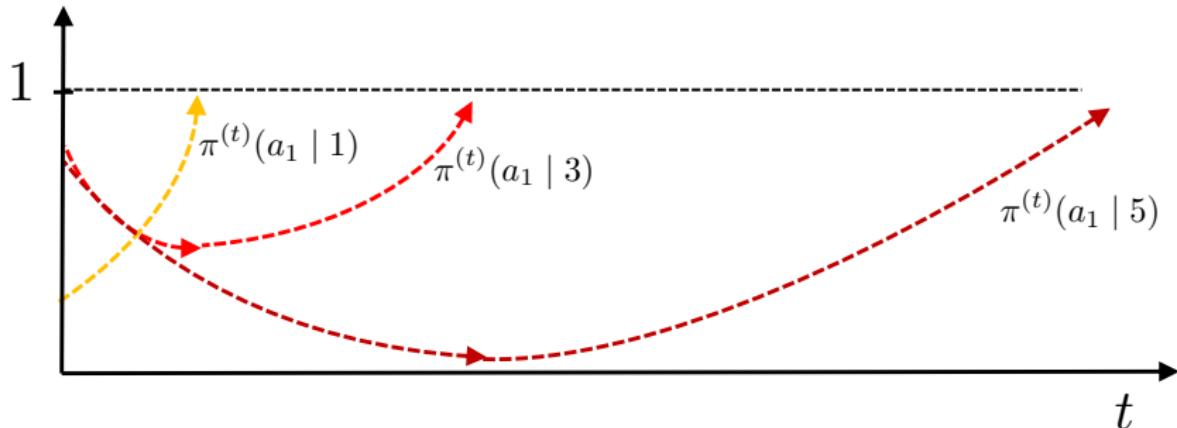


What is happening in our constructed MDP?



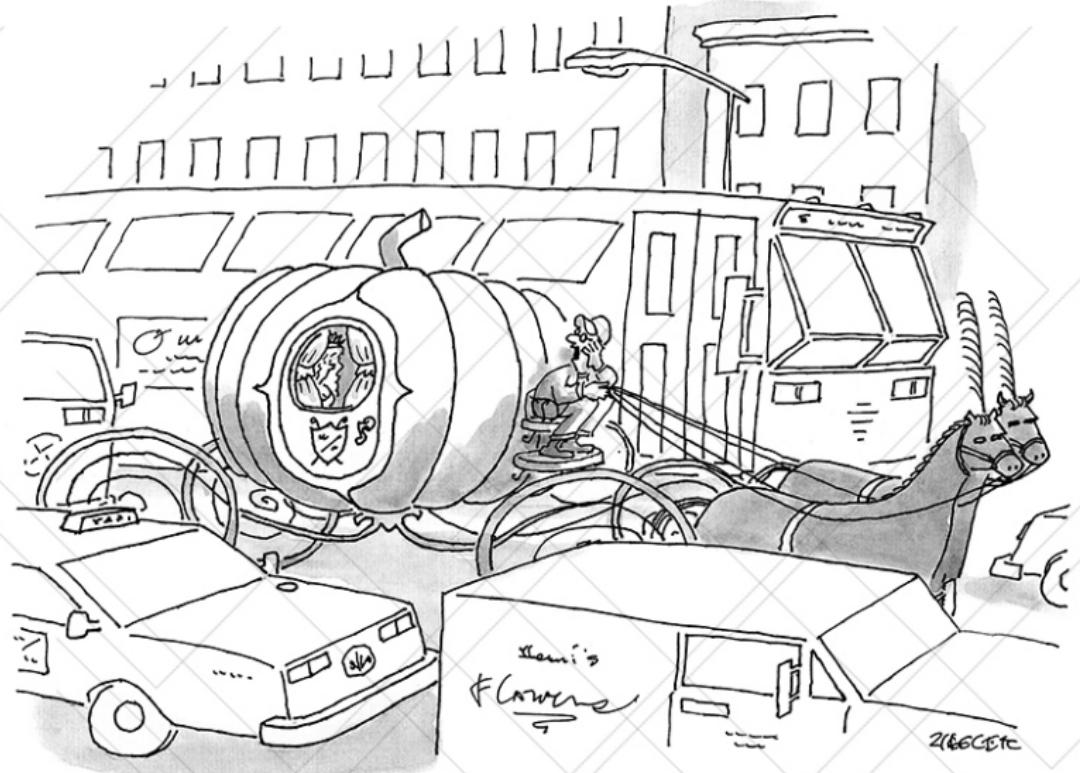
Convergence time for state s grows geometrically as s increases

What is happening in our constructed MDP?



Convergence time for state s grows geometrically as s increases

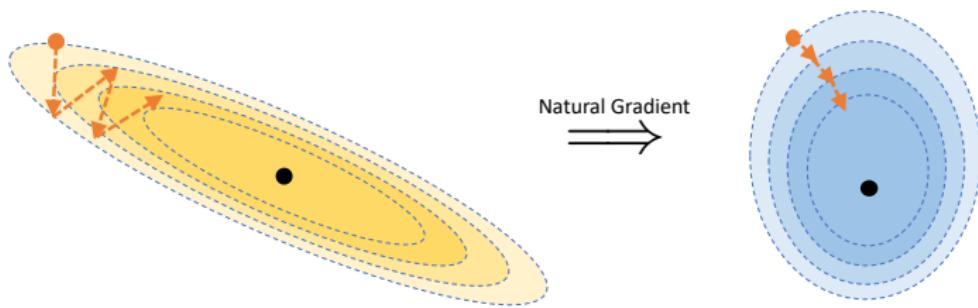
$$\text{convergence-time}(s) \gtrsim (\text{convergence-time}(s - 2))^{1.5}$$



*"Seriously, lady, at this hour you'd make a
lot better time taking the subway."*

Natural policy gradient methods

Natural policy gradient



Natural policy gradient (NPG) method [Kakade, 2001]

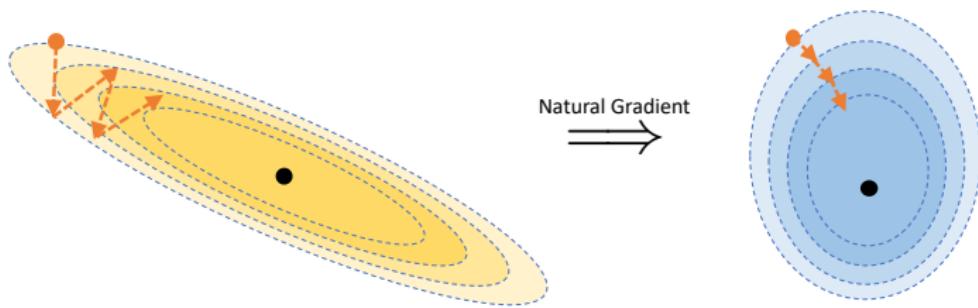
For $t = 0, 1, \dots$

$$\theta^{(t+1)} = \theta^{(t)} + \eta (\mathcal{F}_\rho^\theta)^\dagger \nabla_\theta V^{\pi_\theta^{(t)}}(\rho)$$

where η is the learning rate and \mathcal{F}_ρ^θ is the Fisher information matrix:

$$\mathcal{F}_\rho^\theta := \mathbb{E} \left[(\nabla_\theta \log \pi_\theta(a|s)) (\nabla_\theta \log \pi_\theta(a|s))^\top \right].$$

Natural policy gradient



Natural policy gradient (NPG) method [Kakade, 2001]

For $t = 0, 1, \dots$

$$\theta^{(t+1)} = \theta^{(t)} + \eta (\mathcal{F}_\rho^\theta)^\dagger \nabla_\theta V^{\pi_\theta^{(t)}}(\rho)$$

where η is the learning rate and \mathcal{F}_ρ^θ is the Fisher information matrix:

$$\mathcal{F}_\rho^\theta := \mathbb{E} \left[(\nabla_\theta \log \pi_\theta(a|s)) (\nabla_\theta \log \pi_\theta(a|s))^\top \right].$$

Connection with TRPO/PPO

TRPO/PPO (Schulman et al., 2015; 2017) are popular heuristics in training RL algorithms, with **KL regularization**

$$\text{KL}(\pi_\theta^{(t)} \parallel \pi_\theta) \approx \frac{1}{2} (\theta - \theta^{(t)})^\top \mathcal{F}_\rho^\theta (\theta - \theta^{(t)})$$

via constrained or proximal terms:

$$\begin{aligned}\theta^{(t+1)} &= \operatorname{argmax}_\theta V^{\pi_\theta^{(t)}}(\rho) + (\theta - \theta^{(t)})^\top \nabla_\theta V^{\pi_\theta^{(t)}}(\rho) - \eta \text{KL}(\pi_\theta^{(t)} \parallel \pi_\theta) \\ &\approx \theta^{(t)} + \eta (\mathcal{F}_\rho^\theta)^\dagger \nabla_\theta V^{\pi_\theta^{(t)}}(\rho),\end{aligned}$$

leading to exactly NPG!

Connection with TRPO/PPO

TRPO/PPO (Schulman et al., 2015; 2017) are popular heuristics in training RL algorithms, with **KL regularization**

$$\text{KL}(\pi_\theta^{(t)} \| \pi_\theta) \approx \frac{1}{2} (\theta - \theta^{(t)})^\top \mathcal{F}_\rho^\theta (\theta - \theta^{(t)})$$

via constrained or proximal terms:

$$\begin{aligned}\theta^{(t+1)} &= \operatorname{argmax}_\theta V^{\pi_\theta^{(t)}}(\rho) + (\theta - \theta^{(t)})^\top \nabla_\theta V^{\pi_\theta^{(t)}}(\rho) - \eta \text{KL}(\pi_\theta^{(t)} \| \pi_\theta) \\ &\approx \theta^{(t)} + \eta (\mathcal{F}_\rho^\theta)^\dagger \nabla_\theta V^{\pi_\theta^{(t)}}(\rho),\end{aligned}$$

leading to exactly NPG!

NPG \approx TRPO/PPO!

NPG in the tabular setting

Natural policy gradient (NPG) method (Tabular setting)

For $t = 0, 1, \dots$, NPG updates the policy via

$$\pi^{(t+1)}(\cdot|s) \propto \underbrace{\pi^{(t)}(\cdot|s)}_{\text{current policy}} \underbrace{\exp\left(\frac{\eta Q^{(t)}(s, \cdot)}{1 - \gamma}\right)}_{\text{soft greedy}} \propto \pi^{(t)}(\cdot|s) \exp\left(\frac{\eta A^{(t)}(s, \cdot)}{1 - \gamma}\right)$$

where $Q^{(t)} := Q^{\pi^{(t)}}$ and $A^{(t)} := A^{\pi^{(t)}}$ is the Q /advantage function of $\pi^{(t)}$, and $\eta > 0$ is the learning rate.

- the derivation is left as an exercise; see [Agarwal et al., 2019].

NPG in the tabular setting

Natural policy gradient (NPG) method (Tabular setting)

For $t = 0, 1, \dots$, NPG updates the policy via

$$\pi^{(t+1)}(\cdot|s) \propto \underbrace{\pi^{(t)}(\cdot|s)}_{\text{current policy}} \underbrace{\exp\left(\frac{\eta Q^{(t)}(s, \cdot)}{1 - \gamma}\right)}_{\text{soft greedy}} \propto \pi^{(t)}(\cdot|s) \exp\left(\frac{\eta A^{(t)}(s, \cdot)}{1 - \gamma}\right)$$

where $Q^{(t)} := Q^{\pi^{(t)}}$ and $A^{(t)} := A^{\pi^{(t)}}$ is the Q /advantage function of $\pi^{(t)}$, and $\eta > 0$ is the learning rate.

- the derivation is left as an exercise; see [Agarwal et al., 2019].
- invariant with the choice of ρ

NPG in the tabular setting

Natural policy gradient (NPG) method (Tabular setting)

For $t = 0, 1, \dots$, NPG updates the policy via

$$\pi^{(t+1)}(\cdot|s) \propto \underbrace{\pi^{(t)}(\cdot|s)}_{\text{current policy}} \exp \left(\frac{\eta Q^{(t)}(s, \cdot)}{1 - \gamma} \right) \propto \pi^{(t)}(\cdot|s) \exp \left(\frac{\eta A^{(t)}(s, \cdot)}{1 - \gamma} \right)$$

soft greedy

where $Q^{(t)} := Q^{\pi^{(t)}}$ and $A^{(t)} := A^{\pi^{(t)}}$ is the Q /advantage function of $\pi^{(t)}$, and $\eta > 0$ is the learning rate.

- the derivation is left as an exercise; see [Agarwal et al., 2019].
- invariant with the choice of ρ
- Reduces to policy iteration (PI) when $\eta = \infty$.

Global convergence of NPG

Theorem 4 ([Agarwal et al., 2021])

Set $\pi^{(0)}$ as a uniform policy. For all $t \geq 0$, we have

$$V^{(t)}(\rho) \geq V^*(\rho) - \left(\frac{\log |\mathcal{A}|}{\eta} + \frac{1}{(1-\gamma)^2} \right) \frac{1}{t}.$$

Global convergence of NPG

Theorem 4 ([Agarwal et al., 2021])

Set $\pi^{(0)}$ as a uniform policy. For all $t \geq 0$, we have

$$V^{(t)}(\rho) \geq V^*(\rho) - \left(\frac{\log |\mathcal{A}|}{\eta} + \frac{1}{(1-\gamma)^2} \right) \frac{1}{t}.$$

Implication: set $\eta \geq (1-\gamma)^2 \log |\mathcal{A}|$, we find an ϵ -optimal policy within at most

$$\frac{2}{(1-\gamma)^2 \epsilon} \text{ iterations.}$$

Global convergence of NPG

Theorem 4 ([Agarwal et al., 2021])

Set $\pi^{(0)}$ as a uniform policy. For all $t \geq 0$, we have

$$V^{(t)}(\rho) \geq V^*(\rho) - \left(\frac{\log |\mathcal{A}|}{\eta} + \frac{1}{(1-\gamma)^2} \right) \frac{1}{t}.$$

Implication: set $\eta \geq (1-\gamma)^2 \log |\mathcal{A}|$, we find an ϵ -optimal policy within at most

$$\frac{2}{(1-\gamma)^2 \epsilon} \text{ iterations.}$$

Global convergence at a sublinear rate independent of $|\mathcal{S}|$, $|\mathcal{A}|$!

Key ingredients of the proof

Lemma 5 (Performance difference lemma)

For all policies π, π' and distributions ρ over \mathcal{S} ,

$$V^\pi(\rho) - V^{\pi'}(\rho) = \frac{1}{1-\gamma} \mathbb{E}_{s' \sim d_\rho^\pi} \mathbb{E}_{a' \sim \pi(\cdot | s')} [A^{\pi'}(s', a')].$$

- measures the performance difference for any pairs of policies

Key ingredients of the proof

Lemma 5 (Performance difference lemma)

For all policies π, π' and distributions ρ over \mathcal{S} ,

$$V^\pi(\rho) - V^{\pi'}(\rho) = \frac{1}{1-\gamma} \mathbb{E}_{s' \sim d_\rho^\pi} \mathbb{E}_{a' \sim \pi(\cdot | s')} [A^{\pi'}(s', a')].$$

- measures the performance difference for any pairs of policies

Lemma 6 (Policy improvement of NPG)

$$V^{(t+1)}(\rho) - V^{(t)}(\rho) \geq \frac{(1-\gamma)}{\eta} \mathbb{E}_{s \sim \rho} \log Z_t(s) \geq 0$$

where $Z_t(s) = \sum_a \pi^{(t)}(a|s) \exp(\eta A^{(t)}(s, a)/(1-\gamma))$.

- monotonic performance improvement of NPG

Step 1: bounding the optimality gap

Denote $d^* := d_\rho^*$, and $\pi_s := \pi(\cdot|s)$. By the performance difference lemma,

$$\begin{aligned} & V^*(\rho) - V^{(t)}(\rho) \\ &= \frac{1}{1-\gamma} \mathbb{E}_{s \sim d^*} \sum_a \pi^*(a|s) A^{(t)}(s, a) \\ &= \frac{1}{\eta} \mathbb{E}_{s \sim d^*} \sum_a \pi^*(a|s) \log \frac{\pi^{(t+1)}(a|s) Z_t(s)}{\pi^{(t)}(a|s)} \\ &= \frac{1}{\eta} \mathbb{E}_{s \sim d^*} \left(\text{KL}(\pi_s^* \| \pi_s^{(t)}) - \text{KL}(\pi_s^* \| \pi_s^{(t+1)}) + \sum_a \pi^*(a|s) \log Z_t(s) \right) \\ &= \frac{1}{\eta} \mathbb{E}_{s \sim d^*} \left(\text{KL}(\pi_s^* \| \pi_s^{(t)}) - \text{KL}(\pi_s^* \| \pi_s^{(t+1)}) + \log Z_t(s) \right). \end{aligned}$$

Step 2: telescoping

By the improvement lemma $V^{(t+1)}(\rho) \geq V^{(t)}(\rho)$,

$$\begin{aligned} V^*(\rho) - V^{(T-1)}(\rho) &\leq \frac{1}{T} \sum_{t=0}^{T-1} \left(V^*(\rho) - V^{(t)}(\rho) \right) \\ &= \frac{1}{\eta T} \sum_{t=0}^{T-1} \mathbb{E}_{s \sim d^*} \left(\text{KL}(\pi_s^* \| \pi_s^{(t)}) - \text{KL}(\pi_s^* \| \pi_s^{(t+1)}) + \log Z_t(s) \right) \\ &\leq \frac{1}{\eta T} \mathbb{E}_{s \sim d^*} \text{KL}(\pi_s^* \| \pi_s^{(0)}) + \frac{1}{\eta T} \sum_{t=0}^{T-1} \mathbb{E}_{s \sim d^*} \log Z_t(s), \end{aligned}$$

Step 2: telescoping

By the improvement lemma $V^{(t+1)}(\rho) \geq V^{(t)}(\rho)$,

$$\begin{aligned} V^*(\rho) - V^{(T-1)}(\rho) &\leq \frac{1}{T} \sum_{t=0}^{T-1} \left(V^*(\rho) - V^{(t)}(\rho) \right) \\ &= \frac{1}{\eta T} \sum_{t=0}^{T-1} \mathbb{E}_{s \sim d^*} \left(\text{KL}(\pi_s^* \| \pi_s^{(t)}) - \text{KL}(\pi_s^* \| \pi_s^{(t+1)}) + \log Z_t(s) \right) \\ &\leq \frac{1}{\eta T} \mathbb{E}_{s \sim d^*} \text{KL}(\pi_s^* \| \pi_s^{(0)}) + \frac{1}{\eta T} \sum_{t=0}^{T-1} \mathbb{E}_{s \sim d^*} \log Z_t(s), \end{aligned}$$

where the second term is bounded by the policy improvement lemma

$$\begin{aligned} \frac{1}{\eta} \sum_{t=0}^{T-1} \mathbb{E}_{s \sim d^*} \log Z_t(s) &\leq \frac{1}{1-\gamma} \sum_{t=0}^{T-1} \left(V^{(t+1)}(d^*) - V^{(t)}(d^*) \right) \\ &\leq \frac{1}{1-\gamma} \left(V^{(T)}(d^*) - V^{(0)}(d^*) \right) \end{aligned}$$

Step 3: finishing up

Putting the above together,

$$\begin{aligned} & V^\star(\rho) - V^{(T-1)}(\rho) \\ & \leq \frac{1}{\eta T} \mathbb{E}_{s \sim d^\star} \text{KL}(\pi_s^\star \| \pi_s^{(0)}) + \frac{1}{(1-\gamma)T} \left(V^{(T)}(d^\star) - V^{(0)}(d^\star) \right) \\ & \leq \frac{\log |\mathcal{A}|}{\eta T} + \frac{1}{(1-\gamma)^2 T}, \end{aligned}$$

where we used $\text{KL}(\pi_s^\star \| \pi_s^{(0)}) \leq \log |\mathcal{A}|$ and $V \leq \frac{1}{1-\gamma}$.

Proof of Lemma 6

Proof of $\log Z_t(s) \geq 0$:

$$\begin{aligned}\log Z_t(s) &= \log \sum_a \pi^{(t)}(a|s) \exp\left(\eta A^{(t)}(s, a)/(1 - \gamma)\right) \\ &\geq \sum_a \pi^{(t)}(a|s) \log \exp\left(\eta A^{(t)}(s, a)/(1 - \gamma)\right) \quad (\text{Jensen's inequality}) \\ &= \frac{\eta}{1 - \gamma} \sum_a \pi^{(t)}(a|s) A^{(t)}(s, a) \\ &= \frac{\eta}{1 - \gamma} \sum_a \pi^{(t)}(a|s) (Q^{\pi^{(t)}}(s, a) - V^{\pi^{(t)}}(s)) \\ &= 0\end{aligned}$$

Proof of Lemma 6

Bounding $V^{(t+1)}(\rho) - V^{(t)}(\rho)$: by the performance difference lemma,

$$\begin{aligned} V^{(t+1)}(\rho) - V^{(t)}(\rho) &= \frac{1}{1-\gamma} \mathbb{E}_{s \sim d_\rho^{(t+1)}} \sum_a \pi^{(t+1)}(a|s) A^{(t)}(s, a) \\ &= \frac{1}{\eta} \mathbb{E}_{s \sim d_\rho^{(t+1)}} \sum_a \pi^{(t+1)}(a|s) \log \frac{\pi^{(t+1)}(a|s) Z_t(s)}{\pi^{(t)}(a|s)} \\ &= \frac{1}{\eta} \mathbb{E}_{s \sim d_\rho^{(t+1)}} \text{KL}(\pi^{(t+1)}(s) \| \pi^{(t)}(s)) + \frac{1}{\eta} \mathbb{E}_{s \sim d_\rho^{(t+1)}} \log Z_t(s) \\ &\geq \frac{(1-\gamma)}{\eta} \mathbb{E}_{s \sim \rho} \log Z_t(s), \end{aligned}$$

where we use $d_\rho^{(t+1)} \geq (1-\gamma)\rho$ and $\log Z_t(s) \geq 0$.

References I

-  Agarwal, A., Jiang, N., Kakade, S. M., and Sun, W. (2019).
Reinforcement learning: Theory and algorithms.
-  Agarwal, A., Kakade, S. M., Lee, J. D., and Mahajan, G. (2021).
On the theory of policy gradient methods: Optimality, approximation, and distribution shift.
The Journal of Machine Learning Research, 22(1):4431–4506.
-  Kakade, S. M. (2001).
A natural policy gradient.
Advances in Neural Information Processing Systems, 14.
-  Li, G., Wei, Y., Chi, Y., and Chen, Y. (2023).
Softmax policy gradient methods can take exponential time to converge.
Mathematical Programming.
-  Mei, J., Xiao, C., Szepesvari, C., and Schuurmans, D. (2020).
On the global convergence rates of softmax policy gradient methods.
In *International Conference on Machine Learning*, pages 6820–6829. PMLR.
-  Sutton, R. S., McAllester, D., Singh, S., and Mansour, Y. (1999).
Policy gradient methods for reinforcement learning with function approximation.
Advances in neural information processing systems, 12.